

LASKENTAOHJELMIEN KEHITYS- JA PÄIVITYSTYÖ

Konecranes

Jarno Lahti

Opinnäytetyö
Toukokuu 2013
Kone- ja tuotantotekniikka
Tuotekehitys

TAMPEREEN AMMATTIKORKEAKOULU
Tampere University of Applied Sciences

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Kone- ja tuotantotekniikan koulutusohjelma
Tuotekehitys

JARNO LAHTI

Laskentaohjelmien kehitys- ja päivitystyö

Opinnäytetyö 42 sivua, joista liitteitä 5 sivua
Toukokuu 2013

Opinnäytetyössä kartoitettiin yrityksen käyttöön aikanaan Fortranilla ja C-kielellä ohjelmoitujen laskentaohjelmien tarpeellisuutta, ja niiden päivitys- ja kehittämismahdollisuutta nykyisissä käyttöjärjestelmissä toimiviksi. Yrityksessä nykyisin toimivat laskentaohjelmat avautuvat vain 32-bittisessä käyttöjärjestelmässä ja siellä on siirrytty suuremmilta osin käyttämään 64-bittistä Windows-käyttöjärjestelmää.

Tavoitteena työssä oli selvittää paras päivitysmenetelmä laskentaohjelmille ja saada ne toimimaan 64-bittisessä käyttöjärjestelmässä. Lisäksi tutkittiin mahdollisia kehittämistarpeita laskentaohjelmien käyttöliittymiin. Toteutusmenetelmiä vertailtiin keskenään parhaimman selvittämiseksi. Työn tilaajana toimi nostolaitteita valmistava Konecranes Oy.

Laskentaohjelmien päivitystyö päätettiin tehdä yhteistyönä paikallisen ohjelmointiyrityksen kanssa. Toteutustavaksi päätettiin ohjelmien uudelleen kääntäminen toimivaksi 64-bittisessä käyttöjärjestelmässä ja mahdollisesti ohjelmien käyttöliittymän uudistaminen. Laskentaohjelmat saatiin kumminkin toimimaan ennen tätä 64-bittisessä Windowsissa DOSBox -emulaattorin avulla, joten työn tavoitteeseen päästiin.

Jatkokehitysmahdollisuutena on mahdollisesti laskentaohjelmien sisällyttäminen Konecranesin omaan laskentaohjelma-perheeseen tai täysin oman käyttöliittymän kehittäminen.

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree Programme in Mechanical and Production Engineering
Option of Product Development

JARNO LAHTI

Development and Update Process of Calculation Programs

Bachelor's thesis 42 pages, appendices 5 pages
May 2013

The purpose of this bachelor's thesis was to study the company's old calculation programs and find the best way to develop and update programs to function with Microsoft Windows 64-bit operating system. Konecranes Ltd, the manufacturer of heavy lifting products commissioned the work to me. The thesis was done in Konecranes Port Cranes department in the mechanical designing group.

The study was carried out as a project and covers the Port Cranes department's calculation programs. One of the goals of the thesis was to study whether all the programs are necessary nowadays. The basic idea in the project was to update all the programs that are needed in our department and cut all those that are not. Next step was to find a method to update the calculation programs. Different update methods were compared and finally, the best one was chosen. These results suggest that it was better to choose some subcontractor than let someone in Konecranes do the project.

Finally it was decided that the updating work is done by the local software company NSD. First they work for one or two weeks to compile one of the calculation programs. After the one or two weeks it is known what is the final workload and schedule. If the programs are working properly, then the software company can compile the rest of the programs too.

The future plan is to get all these compiled programs included into the company's own calculation software interface.

Key words: calculation program, programming, update

SISÄLLYS

1	JOHDANTO.....	7
1.1	Opinnäytetyön tarkoitus.....	7
1.2	Taustaa	7
1.3	Tavoite	7
2	KONECRANES OY	8
2.1	Historia.....	8
2.2	Strategia	9
2.3	Tuotteet ja palvelut	10
2.3.1	Siltanosturit	11
2.3.2	Kappaleenkäsittelylaitteet	11
2.3.3	Työpistenostolaitteet	12
2.3.4	Ydinvoimaloiden laitteet.....	13
2.3.5	Satamanosturit.....	13
2.3.6	Trukit.....	15
3	TAUSTAA	16
3.1	Lähtökohdat	16
3.2	Ohjelmat.....	17
3.3	Ohjelmointi	18
3.4	Ohjelmointikielet	19
3.4.1	Fortran	20
3.4.2	BASIC	21
3.4.3	C++.....	21
4	OHJELMISTOSELVITYS	23
4.1	Laskentaohjelmien kartoitus	23
4.2	CTPCMAN	24
4.2.1	Käyttöliittymän kulkukaavio.....	25
4.2.2	Toimintaperiaate	25
5	TOTEUTUSVAIHTOEHDOT	28
5.1	Ohjelmien priorisointi.....	28
5.2	Menetelmävaihtoehdot.....	28
5.2.1	Ohjelmisto kääntäjä.....	28
5.2.2	Excel.....	29
5.2.3	Emulaattori	29
5.2.4	KC -laskentaohjelmat	30
5.2.5	Päivitystyön alihankinta	31
6	TOTEUTUSVAIHTOEHDON VALINTA	32

6.1 Päivitystyön suorittajan valinta.....	33
6.2 Päivitysratkaisu	34
7 POHDINTA JA JATKOKEHITYSMAHDOLLISUUDET	35
7.1 Pohdinta	35
7.2 Jatkokehitysmahdollisuudet.....	36
LÄHTEET.....	37
LIITTEET	38
Liite 1. CTPCMAN –käyttöliittymän kulkukaavio	38
Liite 2. Esimerkki CTPCMAN –käyttöliittymän lähdekoodista	39
Liite 3. Laskentaohjelmien päivityssuunnitelma (NSD) 1(3).....	40
Liite 3. Laskentaohjelmien päivityssuunnitelma (NSD) 2(3).....	41
Liite 3. Laskentaohjelmien päivityssuunnitelma (NSD) 3(3).....	42

ERITYISSANASTO

Excel	Windowsin taulukkolaskentaohjelma.
HP	Hewlett-Packard tietotekniikkavalmistaja.
HP-3000	Tietokonemalli.
CAD	Tietokoneavusteinen suunnittelu.
Emulaattori	Mahdollistaa ohjelmien toiminnan muunlaisella käyttöjärjestelmällä.
EXE	Ohjelmointikielestä käännetty, suorituskelpoinen tiedostomuoto.
BSE	Nosturitiedosto, sisältää kaikki tietyn nosturityypin teknilliset parametrit, eli nosturityyppiin liittyvien laskentaohjelmien alkuarvot selityksineen.
Lähdekoodi	Ohjelmoinnissa tietokoneohjelman tekstimuotoinen ohjelmointikielestä oleva lista.
CTPCMAN	CT -laskentaohjelmien käyttöliittymä, joilla ohjelmia ajetaan.
CT	Laskentaohjelma, esim. CT114, tietty numeroyhdistelmä perässä.
BAS	Laskentaohjelma, esim. BAS45. Ohjelmien alku tulee ohjelmointikielestä Basic.
TDT	Fortran –kielinen laskentaohjelma, jonka CT –laskentaohjelmat ovat korvanneet.
INB	Vanha laskentaohjelma, jonka CT –ohjelmat ovat korvanneet.
BSE	Nosturitiedosto laskentaohjelmassa, joka sisältää kaikki nosturityypille olevat alkuarvot laskennassa.

1 JOHDANTO

1.1 Opinnäytetyön tarkoitus

Opinnäytetyössä kartoitettiin yrityksen vanhojen laskentaohjelmien tarpeellisuutta ja päivitysmahdollisuuksia nykyisille käyttöjärjestelmille. Työn tilaajana toimii Hyvin-
källä pääkonttoriaan pitävä nostolaitteita valmistava Konecranes Oy.

1.2 Taustaa

Tutkintotyön tarve tuli yrityksen satamanosturi suunnitteluyksikön ehdotuksesta päivit-
tää vanhoja 70 - 80 – luvulla tehtyjä laskentaohjelmia sopiviksi nykyisille 64 – bittisille
käyttöjärjestelmille (esim. Windows 7:lle tai uusimmalle Windows 8:lle). Ohjelmat
toimivat nykyisin vain 32-bittisessä käyttöjärjestelmässä, mutta eivät 64-bittisessä. Yri-
tyksessä oli aikaisemmin päivitetty ohjelmia yksi kerrallaan Excel-tiedostoiksi tai kor-
vattu niitä jollain muilla laskentamenetelmillä. Yrityksen toiveena oli, että työssä tutkit-
taisiin miten laskentaohjelmat saataisiin päivitettyä kaikki kerralla ja millä menetelmällä
ne olisi kätevintä päivittää, huomioiden myös, että ne toimisivat myös uusissa tulevilla
käyttöjärjestelmissä.

1.3 Tavoite

Tutkintotyön tavoitteena oli selvittää laskentaohjelmien mahdolliset päivittämismahdol-
lisuudet ja löytää paras päivitysmenetelmä niille. Tarkoituksena oli myös selvittää tar-
peettomaksi jääneitä laskentaohjelmia päivitystyön kannalta priorisoinnin avulla eli lis-
tata ohjelmat tärkeysjärjestykseen. (esim. aiemmin päivitetty tai toisella ohjelmalla kor-
vatut ohjelmat). Priorisointi tapahtui haastattelemalla yrityksessä henkilöitä, jotka las-
kentaohjelmia käyttävät ja heiltä kyselemällä mitkä ohjelmat kannattaisi päivittää.

2 KONECRANES OY

Konecranes Oy on vuonna 1994 syntynyt maailmanlaajuinen nostolaitteita valmistava yhtiö. Yhtiön yksi iso liiketoimi on myös kunnossapito. Kunnossapito huoltaa kaikkia nosturimerkkeitä. Yrityksessä työskentelee maailmanlaajuisesti noin 12100 henkilöä, 626 eri paikassa ja 48 eri maassa. Pääkonttori sijaitsee Hyvinkäällä, jossa työskentelee noin 1200 henkilöä, koko Suomessa työskentelee noin 2000 henkilöä. Vuonna 2012 yrityksen liikevaihto oli 2170 miljoonaa euroa.

2.1 Historia

Konecranes yhtiö perustettiin vuonna 1994, jolloin se irtautui hissivalmistaja KONE yhtiöstä. Konecranes valmistaa nostureita kun taas KONE keskittyy hissien valmistukseen. (Konecranes 2013. Konecranesin historia).

Konecranesin historia ulottuu vuoteen 1910, jolloin perustettiin sähkömoottoreiden korjaamiseen erikoistunut KONE Oy. Vuonna 1933 yhtiö aloitti nykypäivänäkin valmistettavien suurten teollisuusnostureiden tuotannon. Yhtiön tärkeimpiä asiakkaita olivat tuolloin sellu- ja paperi- sekä voimalaitosteollisuudet. (Konecranes 2013. Konecranesin historia).

50-luvulla sotien jälkeen yhtiön satamanostureiden valmistus kasvoi roimasti, talouden parantumisen seurauksena. 70-luvulla yhtiö aloitti kansainvälisen laajentumisen, jolloin se teki ensimmäisen yritysostonsa. Hankittu yhtiö oli Norjalainen Wisbech-Refsum. Yhtiö alkoi tämän jälkeen tehdä enemmänkin yritysostoja ja laajeni näin maailmanlaajuisesti yritykseksi. 80-luvulla yhtiö osti niin Yhdysvaltalaisen R&M Materials Handlingin kuin Ranskalaisen Verlindenin. Yhtiö siis laajeni erityisesti yritysostoillaan. (Konecranes 2013. Konecranesin historia).

80-luvun lopulla yhtiö perusti erillisen nosturidivisioonan, KONE Cranesin (ei siis vielä Konecranes nimellä toimiva). Vuonna 1991 yritys aloitti mittavan rakenneuudistusohjelman, joka kesti aina vuoteen 1994 asti. Lopulta samana vuonna KONE Cranes aloittaa toimintansa itsenäisenä yhtiönä ilman emoyhtiö KONE-konsernia. KCI Konecranes International perustettiin 15.4.1994. (Konecranes 2013. Konecranesin historia).

KCI Konecranes listautui Helsingin pörssiin vuonna 1996. Yhtiö jatkoi yritysostojaan, ostaen yhtiön Saksasta (MAN SWF Krantechnik) ja perustaen tehtaan Kiinan Shanghaihin. Myös Japaniin solmitaan yhteisyrityssopimus. Vuonna 2004 yhtiö lisää tuotevalikoimaansa haarukkatrukit ja kontinkurottajat. (Konecranes 2013. Konecranesin historia).

2.2 Strategia

Strategisena tavoitteena Konecranesilla on säilyttää asemansa maailman suurimpana teollisuusnosturien valmistajana maailmassa. Konecranes haluaa erottua palvelu- ja teknologiainnovaationsa avulla. Yhtiön ja asiakkaan välinen luottamus on erittäin tärkeää. Siksi yhtiö haluaa parantaa kokoajan palvelutarjontaan ja asiakkaan palvelukokemuksia. Yhtiö on pyrkinyt jatkuvasti kasvattamaan tuotekehitysinvestointeja. Turvallisuus ja laatu ovat yhtiölle myös tärkeitä ja yhtiö on sitoutunut samaan ne jokapäiväiseen ajattelutapaan. (Konecranes 2013. Konecranesin strategia).

Konecranes panostaa oman henkilöstönsä kehittämiseen jatkuvasti. Henkilöstön osaaminen ja motivaatio vaikuttavat yrityksen menestykseen. Yhtiö investoi jatkuvasti koulutukseen ja ihmisten johtamistaitojen kehittämiseen. Yhtiö uskoo siihen, että mitä pätevämpiä työntekijät ovat, sitä parempia tuotteita ja palveluita yhtiö tarjoaa. (Konecranes 2013. Konecranesin strategia).

Konecranes haluaa laajentua jatkuvasti. Yhtiö on sitoutunut olemaan aktiivisempi ja maailmanlaajuisempi nosturialan toimija. Yhtiö uskoo, että voisi laajentua ja tarjota tuotteitaan vielä usealle alueelle maailmassa. Yhtiö luottaa pääsevänsä tavoitteisiinsa jatkuvan orgaanisen kasvun ja yritysostojen kautta. (Konecranes 2013. Konecranesin strategia).

Markkinoiden pirstoutuneisuuden vuoksi yhtiö myy tuotteitaan sekä suoraan loppukäyttäjille että pienille itsenäisille nosturivalmistajille ja teollisuuden jakelijoille. Näin yhtiö takaa itselleen mahdollisimman laajan markkina-alueen ja yhtenäisen tuotealustojen tuomat skaalaedut. (Konecranes 2013. Konecranesin strategia).

Yhtiön maailmanlaajuinen läsnäolo ja sitoutuneisuus palvelemaan asiakkaita siellä, missä he ovat edellyttävät globaalia toimitusketjua, joka tarjoaa samanlaiset mahdollisuudet kaikkialla maailmassa ja optimaalisen skaalattavuuden. Yhtiön toimitusketjun on myös pystyttävä sopeutumaan kysynnän vaihteluihin nopeasti, mutta ylläpitämään kilpailukykyiset läpimeno- ja toimitusajat kasvattamatta varastoja. (Konecranes 2013. Konecranesin strategia).

Yhtiö panostaa myös reaaliaikaiseen informaatioon. Päätökset pitää syntyä nopeasti ja perustua tarkkaan tietoon. Yhtiö uskoo hajautettuun päätöksentekoon ja siihen, että päätökset on hyvä tehdä lähellä asiakasta. On ensisijaisen tärkeää, että yhtiö tarjoaa henkilöstölle avointa ja ajan tasalla olevaa tietoa. Yhtiö investoi kokoajan enemmän mahdollisuuksia tarjoaviin uuden sukupolven tietojärjestelmiin. (Konecranes 2013. Konecranesin strategia).

2.3 Tuotteet ja palvelut

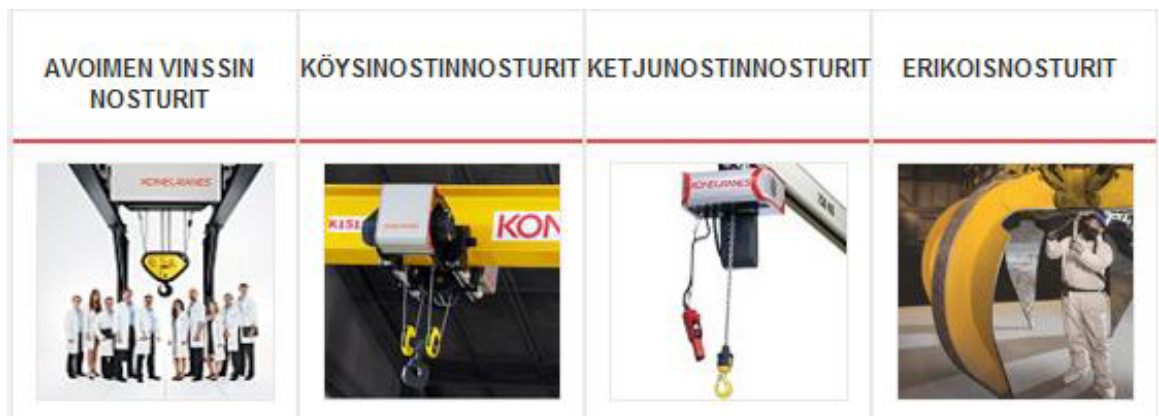
Yrityksen tuotteita ovat siis erilaiset nostolaiteratkaisut. Tuotteisiin kuuluvat siltanosturit, työpistenostotuotteet, kappaleenkäsittelylaitteet, ydinvoimaloidenlaitteet, satamanosturit ja trukit. Myös kunnossapidolla on iso osa Konecranesin liiketoimintaa. Yritys tekee huoltosopimuksia ja huoltaa kaikenmerkkisiä nostolaitteita. Yhtiön palveluita ovat:

- Kunnossapito.
- Konsultointipalvelut.
- Etäpalvelut.
- Modernisoinnit.
- Käyttäjäkoulutus.
- Työstökonehuolto.
- Varaosat.
- Kunnossapitoratkaisut.

2.3.1 Siltanosturit

Siltanostureissa nostin kulkee kahden radan väliin asennettua siltaa pitkin. Nosturit on suunniteltu keskiraskaan ja raskaan teollisuusunostamisen vaatimukset kattaen kaikki valmistusprosessin vaiheet. Siltanostureita on erilaisia ja eri tarkoitukseen soveltuvia (kuva 1). Niitä ovat:

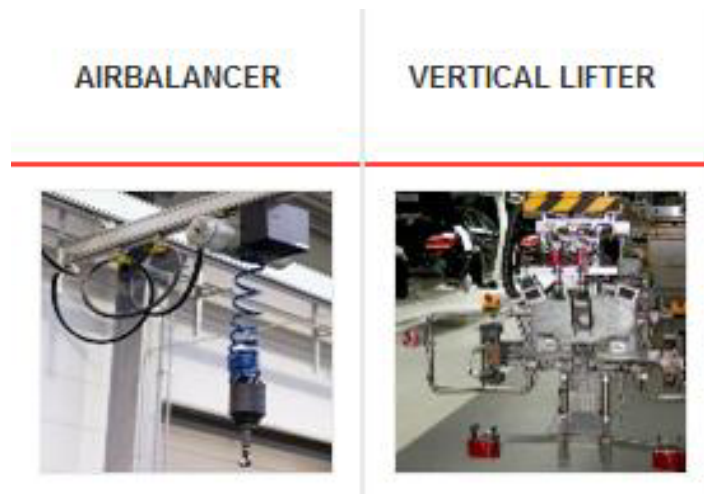
- Avoimen vinssin nosturit.
- Köysinostinnosturit.
- Ketjunostinnosturit.
- Erikoisnosturit.
- Räjähdysvaarallisen toimintaympäristön nosturit.



KUVA 1. Siltanosturit (Konecranes Oy 2013.)

2.3.2 Kappaleenkäsittelylaitteet

Konecranesilla on kaksi tuotetta kappaleenkäsittelyä varten (kuva 2). AirBalancer ja Vertical lifter auttavat tavaroiden manuaalisessa nostamisessa, siirtämisessä ja sijoittamisessa. Niiden avulla voidaan osia ja helposti hajoavia tuotteita siirtää helposti ja nopeasti.

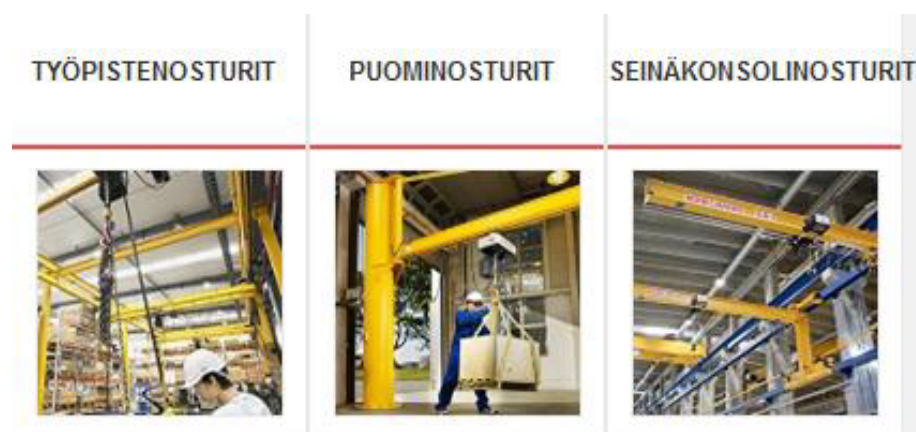


KUVA 2. Kappaleenkäsittelylaitteet (Konecranes Oy 2013.)

2.3.3 Työpistenostolaitteet

Työposteiden nostovaatimuksiin sisältyy helppo asennus ja käyttö. Tärkeitä ovat myös noston tasaisuus ja vankkuus. Konecranesilla on nostolaitteita, jotka käyvät työposteiden kaikkiin materiaalinkäsittelytarpeisiin (kuva 3). Konecranesin työpistenostotuotteita ovat:

- Työpistenosturit.
- Puominosturit.
- Seinäkonsolinosturit.



KUVA 3. Työpistenostolaitteet (Konecranes Oy 2013.)

2.3.4 Ydinvoimaloiden laitteet

Konecranesilla on laaja valikoima sertifioituja ydinvoimalanostureita ja materiaalinkäsittelylaitteita sekä palveluita ydinvoimateollisuudelle. Tuotteet ovat suunniteltu niin, että käyttökatkot vähenevät minimiin ja tuottavuus paranee.

Ydinvoimalateollisuudessa raskasnostolaitteiden on pystyttävä toimimaan erittäin vaativissa olosuhteissa ilman vahinkoja työntekijöille tai ympäristölle. Näissä olosuhteissa ei saa tapahtua voimalan laitteiden tai toimintojen vaarantumista tai kuormanhallinnan menettämistä.

Konecranesin ydinvoimalalaitteet on suunniteltu täyttämään ja ylittämään kaikki vaatimukset, jotka koskevat ydinvoimalateollisuutta. Näitä vaatimuksia ovat eri teollisuuteen liittyvät standardit. Konecranesin ydinvoimalatuotteita (kuva 4) ovat:

- Polar-nosturi.
- Polttoainenkäsittelynosturi.
- Tynnyrikuljetin.
- Turbiinihallinosturi.
- Tynnyrinosturi.



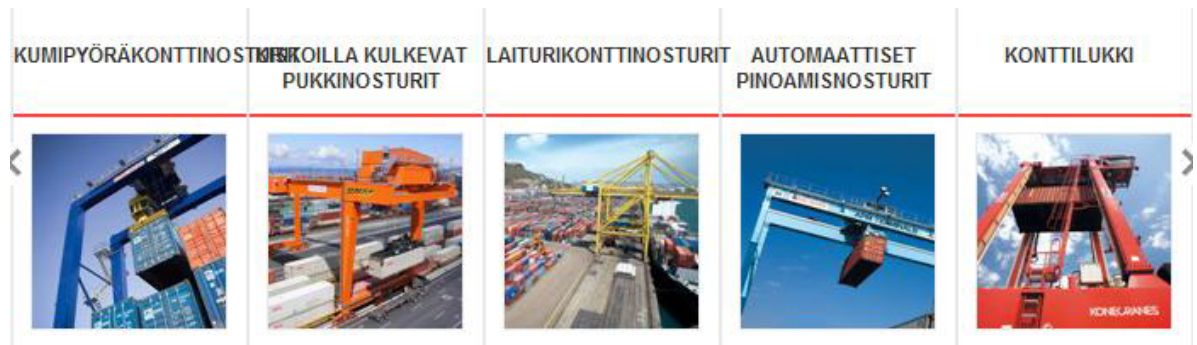
KUVA 4. Ydinvoimaloiden laitteet (Konecranes Oy 2013.)

2.3.5 Satamanosturit

Konecranes toimittaa monipuolisesti paljon eri satamanosturituotteita ympäri maailmaa. Minä itsekkin teen oman opinnäytetyöni satamanosturipuolella. Yhtiö on merkittävä

maailmanlaajuinen toimija kontinkäsittelylaitteiden suunnittelussa, valmistuksessa, toimituksessa ja kunnossapidossa. Konecranesin satamanosturituotteita (kuva 5) ovat:

- Kumipyöräkonttinosturit (RTG).
- Kiskoilla kulkevat pukkinosturit (RMG).
- Laiturikonttinosturit (STS).
- Automaattiset pinoamisnosturit (ASC).
- Konttilukki.
- BOXRUNNER.
- Yard IT konttien käsittelyyn.
- AGD-kahmaripukkinosturi.
- Telakkapukkinosturi.



KUVA 5. Satamanosturit (Konecranes Oy 2013.)

2.3.6 Trukit

Konecranesilta lähti ensimmäinen trukki asiakkaalle 1950-luvulla. Nykyään yhtiö on maailman johtava, nostokapasiteetiltaan 10–80 tonnin trukkien valmistaja. Konecranesin trukkilikoimassa (kuva 6) ovat:

- Haarukkatrukit.
- Kontinkurottajat.
- Tyhjien konttien käsittelylaitteet.
- Pukkinosturit/Toploaderit.



HAARUKKATRUKIT



KONTINKUROTTAJAT



TYHJEN KONTTIEN
KÄSITTELYLAITTEET



PUKKINOSTURIT/TOPLOADERIT

KUVA 6. Trukit (Konecranes Oy 2013.)

3 TAUSTAA

Opinnäytetyössä kartoitettiin satamanosturisuunnittelussa käytettyjen, aikanaan Fortranilla ja C –ohjelmointikielillä ohjelmoituja ja Exe –tiedostoiksi käännettyjen laskentaohjelmien käyttötarkoitukset. Ohjelmia käytetään suurimmaksi osaksi tuoteryhmissä lujuuslaskennassa ja koneistolaskennassa. Suurin osa ohjelmista oli tarkoitettu nosturin rakenteen ja koneiston mitoittamiseen. Ohjelmilla lasketaan muun muassa sallitut kuormitukset ja jännitykset eri tilanteissa. Näiden tietojen avulla pystytään valitsemaan haluttu koneisto tai mitoittamaan haluttu osa nosturiin. Laskentaohjelmista listattiin tarpeelliset ja tarpeettomat. Tarpeettomilla ohjelmilla tarkoitetaan ohjelmia, jotka eivät tarvitse päivitystä, esim. ohjelmat, jotka ovat jo päivitettyjä tai ne ovat korvattu muilla laskentamenetelmillä. Karsinta tehtiin haastattelemalla ohjelmien käyttäjiä. Tämän jälkeen tiedettiin päivitystä tarvitsevat laskentaohjelmat.

Laskentaohjelmat ovat aikanaan tehty helpottamaan nosturien suunnittelutyötä. Niiden ansiosta ei tarvitse joka kerta etsiä oikeaa kaavaa taulukkokirjasta ja sitä kautta koittaa analysoida tai mitoittaa tiettyä osaa projektia varten. Laskentaohjelmien avulla syötetään arvot ja ohjelma laskee sallitut kuormitukset. Näiden tietojen avulla voidaan mitoittaa nosturin osat ja aikaa säästyy.

Ongelmana ohjelmissa on se etteivät ne toimi enää 64-bittisessä käyttöjärjestelmässä, jota käytetään suurimmassa osassa osaston tietokoneista. Vain harvoissa koneissa on 32-bittinen järjestelmä (kannettavissa), jossa laskentaohjelmat toimisivat. Ratkaisuna on laskentaohjelmien uudelleen kääntäminen 64-bittisessä käyttöjärjestelmässä toimiviksi, hankkia sopiva emulaattori tai toteuttaa vastaavat toiminnot Windows Excel –taulukkolaskentaohjelmalla tai liittää ne osaksi Konecranesin omaa ohjelmistoperhettä.

3.1 Lähtökohdat

Tämän työn aloitushetkellä tutkintotyön tekijällä ei ollut aikaisempaa kokemusta laskentaohjelmista eikä ohjelmoinnista. Kaikki oli uutta lähdettäessä tekemään opinnäytetyötä. Työpaikka oli ennestään tuttu edelliskesäisestä kesäharjoittelupaikasta.

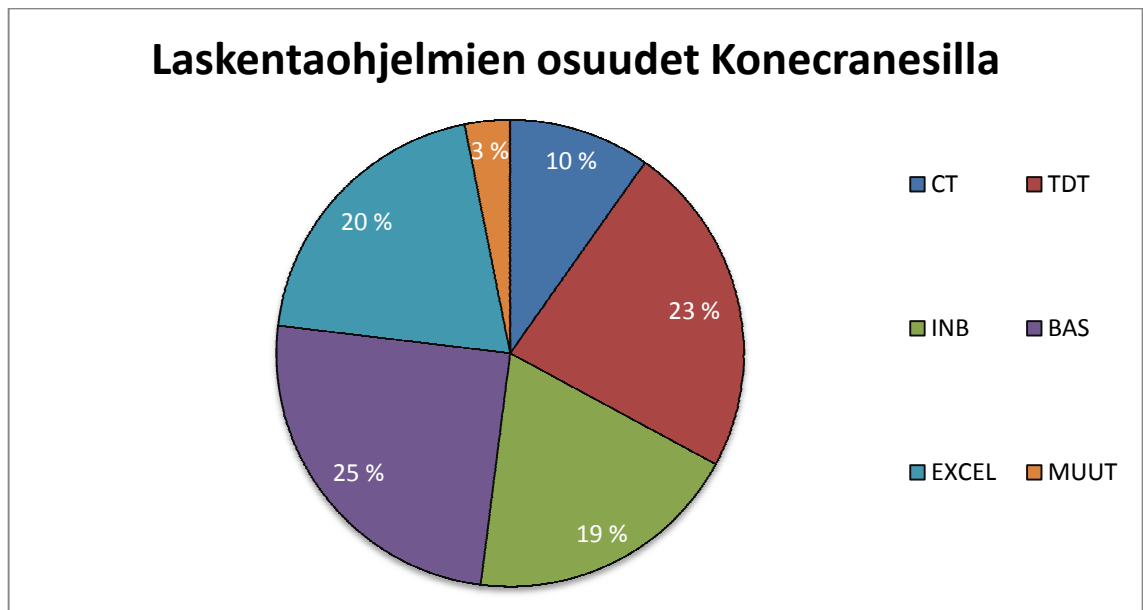
3.2 Ohjelmat

Satamanosturipuolella oli kaiken kaikkiaan noin 225 laskentaohjelmaa ja näiden lisäksi vielä yrityksen omia KC -laskentaohjelmia. Näistä suurin osa oli vanhoja tarpeettomaksi jääneitä ohjelmia, jotka oli korvattu jo muilla ohjelmilla tai niiden antamaa laskentatietoa ei enää tarvittu. Eräät laskentaohjelmat toimivat, mutta ne antoivat vääriä tuloksia ja sen takia ne oli korvattu tai jätetty tarpeettomaksi. Käytetyimmät ja uusimmat laskentaohjelmat olivat CTPCMAN –laskentaohjelmia. CTPCMAN on laskentaohjelmien käyttöliittymä, josta itse CT -laskentaohjelmat valitaan nosturityypin mukaan. Laskentaohjelmat täytyy aina avata CTPCMAN –käyttöliittymän kautta, jotta niitä voisi käyttää. Toisin sanoen CTXXXPC on itse laskentaohjelma ja CTPCMAN käyttöliittymä, jolla ohjelma avataan. Alla joitain esimerkkejä CT -laskentaohjelmista:

- Nosturin vinoonajon voimat.
- Jäykkäpuominosturin kääntökoneisto.
- Jäykkäpuominosturin ratakäyrät.
- Nivelpuominosturin säteiskoneiston moottorit.
- Kahmaripukkinosturin kapasiteetti.
- Nosturin kaarreajon laskenta.
- Nosturin kuljetus ponttonilla.
- Puominostokoneiston moottorin laskelma.

Myös BAS –laskentaohjelmilla on oma käyttöliittymänsä, jonka kautta ne käynnistyvät. Käyttöliittymän nimi on QB (QuickBasic). Se toimii samaan tapaan kuin CTPCMAN –käyttöliittymä CT -laskentaohjelmilla. Ensin avataan QB –käyttöliittymä ja sen kautta valitaan haluttu BAS –laskentaohjelma. BAS –laskentaohjelmat kuuluvat yhtiön vanhempiin laskentaohjelmiin, joista osa jo korvattu CT –laskentaohjelmilla.

Kuviosta (kuvio 1) nähdään esimerkiksi CT –laskentaohjelmien osuus kaikista satamanosturipuolella käytetyistä laskentaohjelmatyypeistä. Suurin osa päivitettävistä ohjelmista on CT –ohjelmia ja muutama erillinen ohjelma, joten päivitettävien ohjelmien osuus kaikista on noin 10%. CT –laskentaohjelmat ovat korvanneet suurimman osan vanhoista TDT ja INB laskentaohjelmista, joita ei nykyisin enää tarvita.



KUVIO 1. Laskentaohjelmajärjestelmien osuudet kaikista laskentaohjelmista Konecranesin satamanosturiyksikössä

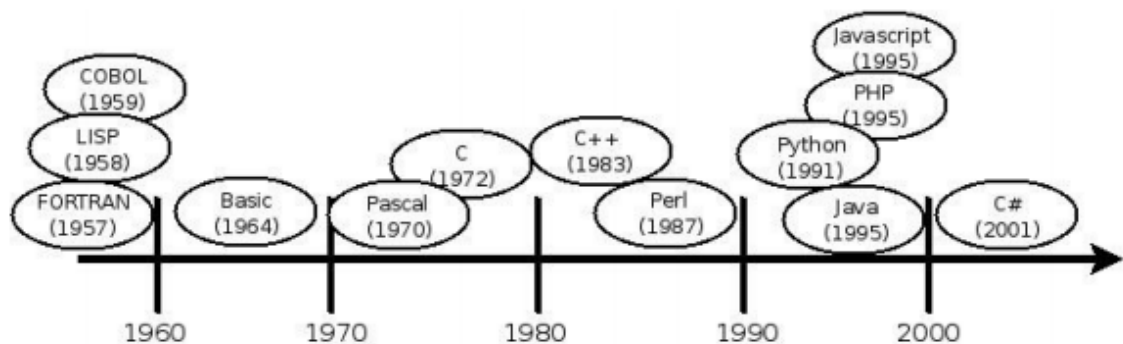
3.3 Ohjelmointi

Yksinkertaisimmin ilmaistuna ohjelmointi on sitä, että ohjelmoija kirjoittaa käskyjä, jotka tietokone tulkitsee ja joiden perusteella tietokone lopulta tekee jotakin. Tietokoneohjelmassa määritellään kaikki yksinkertaisimmatkin toimenpiteet, jotka tietokoneen halutaan tekevän, oli se sitten keltaisen pallon piirtäminen ruudulle tai vaikka vain seuraavan käskyn odottaminen. Ohjelmakoodin tulee olla virheetöntä ja yksiselitteistä, sillä tietokone ei pysty arvailemaan mitä ohjelmoija kenties tarkoitti, siten ei myöskään korjaamaan koodia oikeanlaiseksi. Viallinen koodi voi johtaa siihen, ettei ohjelma toimi halutulla tavalla. (Beginning Programming, Kingsley-Hughes & Kingsley-Hughes 2005)

Ihmisen kirjoittama koodi tulee ensin kääntää tai tulkata konekielelle, ennen kuin tietokone pystyy ymmärtämään sitä. Konekieli koostuu biteistä eli ykkösistä ja nolista. Jotkin ohjelmointikielet, kuten Visual Basic, vaativat oman kehitysympäristön, kun taas toisilla kielillä voi tehdä ohjelmia ihan tavallisella tekstinkäsittelyohjelmalla. (Beginning Programming, Kingsley-Hughes & Kingsley-Hughes 2005)

3.4 Ohjelmointikielet

Ohjelmointia ja ohjelmointikieliä käytetään tietokoneen toimintojen ohjaamisissa. Satamanosturipuolen laskentaohjelmien ohjelmointikieliä olivat ainakin Fortran, BASIC ja C++. Yhtenä kielenä oli myös HP:n oma ohjelmointikieli, HP-3000 koneille tehty, mutta tämä kieli on ollut viimeksi käytössä yrityksessä noin 20 vuotta sitten. Näitä vanhoja laskelmaohjelmia on jäänyt arkistoihin, eikä niille ole tehty mitään. Näillä ohjelmointikielillä ohjelmoidut ohjelmat olisi tarkoitus päivittää toimiviksi 64-bittisessä Windows 7 käyttöjärjestelmässä. Alla olevassa aikajanassa on kuvattu merkittävimpien ohjelmointikielten syntyminen ja julkaisemisvuodet.



KUVIO 2. Merkittävimpien ohjelmointikielten aikajana (Joentakanen 2010.)

Ohjelmointikielten suosiota toisiinsa voi vertailla eri suosituimmuusindekseistä. Yksi tällainen on TIOBE ohjelmistoyhtiön tekemä (taulukko 1). Indeksi päivittyy kuukausittain ja siitä näkee eri ohjelmointikielten suosituimmuusjärjestyksen. Taulukossa on myös ohjelmien edellisvuoden sijoitukset ja osuudet kaikista kielistä. Eli siitä voi vertailla kuinka paljon ohjelmointikielen suosio on kasvanut tai laskenut vuoden aikana. Taulukon tieto perustuu hakukoneilla kerättyyn tietoon. Tällaisia hakukoneita voi olla mm. Google, Bing, Yahoo! tai Amazon. Hakukoneet keräävät tietoa vaikka siitä kuinka paljon kielillä on käyttäjiä tai kuinka paljon niistä järjestetään mahdollisia opetuskursseja. Taulukko ei kerro sitä mikä ohjelmista olisi paras, vain suosituimman. Seuraavassa oleva TIOBE yhtiön taulukko (taulukko 1) on helmikuulta 2013.

TAULUKKO 1. Ohjelmointikielien suosituimmuusindeksi. (Tiobe 2013.)

Sijoitus helmi.13	Sijoitus helmi.12	Sijoituksen suunta	Kieli	Osuus helmi.13	Muutos helmi.12	Tila
1	1	=	Java	18.387%	+1.34%	A
2	2	=	C	17.080%	+0.56%	A
3	5	↑↑	Objective-C	9.803%	+2.74%	A
4	4	=	C++	8.758%	+0.91%	A
5	3	↓↓	C#	6.680%	-1.97%	A
6	6	=	PHP	5.074%	-0.57%	A
7	8	↑	Python	4.949%	+1.80%	A
8	7	↓	(Visual) Basic	4.648%	+0.33%	A
9	9	=	Perl	2.252%	-0.68%	A
10	12	↑↑↑	Ruby	1.752%	+0.19%	A
11	10	↓	JavaScript	1.423%	-1.04%	A
12	16	↑↑↑↑↑	Visual Basic .NET	1.007%	+0.21%	A
13	13	=	Lisp	0.943%	+0.04%	A
14	15	↑	Pascal	0.932%	+0.12%	A
15	11	↓↓↓	Delphi/Object Pascal	0.886%	-1.08%	A
16	14	↓↓	Transact-SQL	0.773%	-0.07%	A--
17	75	↑↑↑↑↑↑↑	Bash	0.741%	+0.61%	A--
18	26	↑↑↑↑↑↑↑	MATLAB	0.648%	+0.15%	B
19	24	↑↑↑↑↑	Assembly	0.640%	+0.12%	B
20	19	↓	Ada	0.631%	0.00%	B

3.4.1 Fortran

Osa yrityksen laskentaohjelmista on tehty Fortranilla (FORMula TRANslator). Fortran on yksi vanhimmista ja eniten käytetyistä ohjelmointikielistä maailmanlaajuisesti. IBM kehitti ohjelman valmiiksi vuonna 1957 ja sitä pidetään ensimmäisenä korkean tason ohjelmointikielenä. Kielen historia ulottuu IBM:llä vuoteen 1954. Se on myös yleisin Konecranesilla laskentaohjelmissa käytetyistä ohjelmointikielistä. Käytetyimpiä ohjelmia ovat olleet FORTRAN 4,66 ja 77. Fortran oli myös ensimmäinen kieli, josta tehtiin standardi vuonna 1966. Fortranin uusin standardi on vuodelta 2003 ja kehitteillä on vielä uudempi, nimeltään Fortran 2008.

Fortranin tärkeimpiä ominaisuuksia olivat tulosten ja syötön muotoilu, looginen if-lause ja toistorakenne laskurimuuttujan suhteen do-lausekkeella. Kieli on tarkoitettu erityisesti raskaaseen ja tehokkaaseen numeeriseen laskentaan. Fortran kieleen liittyy myös

usein taulukoita ja matriiseja. Ohjelmaa on myös helppo siirtää uudelle koneelle standardoidun kielen avulla. Uusimmilla Fortran-kääntäjillä voidaan kääntää myös vanhempien versioiden ohjelmia.

3.4.2 BASIC

BASIC-ohjelmointikieli (Beginner's All purpose Symbolic Instruction Code) on kehitelty vuonna 1964. Se tehtiin ohjelmoinnin alkeiden opetusta varten. Ohjelmointikieli oli suosittu etenkin 80-luvulla, melkein jokaisella kotitaloudella oli kotikoneessaan BASIC-tulkki. Se oli usein sisäänrakennettuna valmiiksi kotikoneisiin. Tänä päivänä tunnetuin BASIC on Microsoftin Visual Basic. Käytetyimmät BASIC-versiot tietokoneissa ovat suuremmilta osin Microsoftin tekemiä ja tulevat Altair BASIC-tulkista. Altair BASIC oli ensimmäinen Microsoftin julkaisema tietokoneohjelma vuonna 1975. Ensimmäinen BASIC-ohjelma, jossa ei tarvinnut käyttää rivinumeroita oli QuickBASIC-kääntäjä, jota myös Konecranesin BAS -laskentaohjelmissa käytetään käyttöliittymänä. Siitä on myös riisuttu versio ilman kääntäjää, QBasic.

Nykyään Visual Basicin lisäksi tunnettuja BASIC-ohjelmointikieliä ovat REALbasic sekä CoolBasic, DarkBasic ja BlitzBasic. Kolme viimeisintä näistä oli kehitelty erityisesti tietokonepelien kehitykseen. BlitzBasic-ohjelmaa käytettiin muun muassa tunnetun Worms-pelin kehittämiseen.

3.4.3 C++

C++-ohjelmointikieli kehittyi sen edeltäjästä C-ohjelmointikielestä. Se julkaistiin vuonna 1983. C++ on oliopohjainen kieli ja siinä on erona edeltäjäänsä olio-ohjelmointiin ja geneerisyyten liittyviä ominaisuuksia. Kielet olivat proseduraalisia ja niissä yksi toiminto luki näppäimen painalluksen ja toinen siirsi tietoa paikasta toiseen. Tämä käytäntö ei toiminut enää isommissa ohjelmissa, joten kehiteltiin oliopohjaisuus. Olio esittää ohjelmaan liittyvää kokonaisuutta, ei vain yhtä toimintoa. C++-ohjelmointikielestä on tehty myös standardi ISO/IEC 14882:1998 vuonna 1998. Uusin standardi kielestä on vuodelta 2011.

Kieli sisältää luokat, periytymisen, mallit ja poikkeukset. C++-ohjelmointikieli kehittyi C-kielestä lisäämällä C-kieleen tiettyjä ominaisuuksia.

C++-ohjelmointikielellä on tehty enemmistö kaikista käyttöjärjestelmä-, ohjelmisto-, verkko- ja peliohjelmista. Näitä ovat muun muassa Microsoftin ja Adobe Systemsin ohjelmat.

Konecranesilla käytössä olevien CT –laskentaohjelmien käyttöliittymän CTPCMANin ohjelmointikieli on juuri C++. Esimerkki ohjelmointikielestä on liitteessä 3. Siinä osa CTPCMAN:n lähdekoodista.

4 OHJELMISTOSELVITYS

4.1 Laskentaohjelmien kartoitus

Aluksi laskentaohjelmista kartoitettiin yrityksen vanhoista kansioista löytyneet ohjelmat. Suurin osa kansioista löytyneistä laskentaohjelmista oli poistunut käytöstä tai korvattu uusilla ohjelmilla. Tehtävänä oli listata jokainen laskentaohjelma, oli se käytössä tai ei. Näiden jälkeen ohjelmia etsittiin verkkolevyiltä. Sieltä löytyi uudempia käytössä olevia ohjelmia, mutta myös vanhoja käytöstä poistuneita. Kaikki löytyneet ohjelmat listattiin Windows Excel-taulukoon (TAULUKKO 2).

Ohjelmistokartoituksen jälkeen tutkittiin erikseen yksittäisen ohjelman käyttötarkoitus. Joissakin kansioissa oli käyttötarkoitus selvitetty, mutta joidenkin ohjelmien kohdalla ohjelma oli käynnistettävä ja selvitettävä sen kautta mitä ohjelma laskee. Kaikki ohjelmat laskivat suurimmaksi osaksi nosturiin tulevia kuormituksia eri käyttötilanteissa.

Laskentaohjelmien ohjelmointikielet täytyi myös selvittää. Ohjelmointikielet selvitettiin katselemalla laskentaohjelmien kansioista löytyviä tietoja ja työkollegoita haastattele-malla. Kartoitustyön jälkeen voitiin arvioida tarvittavan käännöstyön työmäärä ja työmenetelmä.

Kaikista laskentaohjelmista selvitettiin myös ohjelmien lähdekoodit, ilman niitä ohjelmoijan olisi vaikea aloittaa ohjelmien kääntäminen uudelle käyttöjärjestelmälle sopivaksi.

Yritykseen oli tullut uusia ohjelmia, ja vanhojakin oli päivitetty aikaisemminkin. Tutkintotyössä selvitettiin vanhojen ohjelmien käyttötarve. Ohjelma oli mahdollisesti korvattu muulla laskentamenetelmällä tai poistunut kokonaan tarpeettomana käytöstä. Selvityksen yhteenveto on esitetty Excel-taulukkona (TAULUKKO 2).

TAULUKKO 2. Osa laskentaohjelmista Excel-tilukkona.

	A	B	C	D	E
1	OHJELMA	KUVAUS	KIELI	KOMMENTTI	NOSTURITIEDOT
2					
3	CT114PC	Nosturin vinoonajon voimat	FORTTRAN	KCSKEW, EXCEL	SINGLE/DOUBLE/GRABUNLR
4	CT505PC	Jäykkäpuominosturin kääntökoneisto	FORTTRAN	EXCEL, (KC Lift/Drive)	SINGLE
5	CT506PC	Jäykkäpuominosturin säteiskoneisto	FORTTRAN	(KC Lift/Drive)	SINGLE
6	CT707PC	Kääntönosturin kapasiteetti	FORTTRAN	(KC Storage)	SINGLE/DOUBLE
7	CT1008PC	Kääntölaakerin päällinen ja alusta	FORTTRAN	(KC Gantry)	SINGLE/DOUBLE
8	CT2008PC	Jäykkäpuominosturin laskelmat	FORTTRAN	(KC Gantry)	SINGLE
9	CT4002PC	Jäykkäpuominosturin ratakäyrä	FORTTRAN	(KC Rope)	SINGLE
10	CT4002AU	Jäykkäpuominosturin apunoston ratakäyrä	FORTTRAN	(KC Rope)	SINGLE
11	CT4009PC	Jäykkäpuominosturin säteisnopeus	FORTTRAN		SINGLE
12	CT507PC	Nivelpuominosturin säteiskoneiston moottorit	FORTTRAN	(KC Lift/Drive)	DOUBLE
13	CT510PC	Nivelpuominosturin kääntökoneiston moottorit	FORTTRAN	(KC Lift/Drive)	DOUBLE
14	CT4001PC	Nivelpuominosturin vastapaino	FORTTRAN		DOUBLE
15	CT4003PC	Nivelpuominosturin ratakäyrä	FORTTRAN	EXCEL	DOUBLE
16	CT4007PC	Nivelpuomi-konttinosturin kärkipyörästön ratakäyrät	FORTTRAN		DOUBLE
17	CT4008PC	Nivelpuominosturin kuorman punnitus KU:n köysipyörällä	FORTTRAN	(KC Rope)	DOUBLE
18	CT4010PC	Nivelpuominosturin apunoston ratakäyrä	FORTTRAN		DOUBLE
19	CT705PC	Kahmaripukkinosturin kapasiteetti	FORTTRAN	(KC Storage)	GRABUNLR
20	CT708PC	Nivelöityjen vetosauvojen riippuma	FORTTRAN	EXCEL? FINNSAP	CONTRCRN/GRABUNLR
21	CT4005PC	Ulokepukin vaunun siirron köysisovitus	FORTTRAN	(KC Rope)	CONTRCRN/GRABUNLR
22	CT6003PC	Puominostokoneiston moottorin laskenta	FORTTRAN	(KC Lift)	CONTRCRN/GRABUNLR
23	TRIM	Konttinosturin trim-, list- ja skew -koneistot	FORTTRAN	Mahd.Liik.KC Rope	CONTRCRN
24	CT709PC	Konttipukkinosturin kapasiteetti	FORTTRAN	(KC Storage)	CONTRCRN
25	CT408	Kaarrejo laskenta	FORTTRAN		
26	CT905	Nosturin kuljetus ponttoonilla	FORTTRAN		
27	CCPM	Puominostokoneiston laskelma		(KC Gantry)	
28	TDT 350	Nivelpuominosturin laskelmat	FORTTRAN	(KC Gantry)	
29	TDT 511	Nosto- ja siirtomoottoreiden kuormitukset	FORTTRAN	EXCEL	(KC Lift/Drive)
30	TDT 509	Nostokoneiston köysisovitus, tela, välitykset ja niiden momentit	FORTTRAN	EXCEL	
31	BAS45	Telan päädyn väsymislajuuden laskenta		QB45.EXE	Drum05b
32	KON2	Hammaspyöräparin jännitykset	HP-3000 bas	KC Gear, KissSoft	

4.2 CTPCMAN

Konecranes käyttää satamanosturipuolella CT -laskentaohjelmissa CTPCMAN –käyttöliittymää. Käyttöliittymän avulla eri CT –laskentaohjelmat pystytään avaamaan. CT –laskentaohjelmia ei voi avata yksittäin ilman CTPCMAN –käyttöliittymää. Laskentaohjelmat tehdään FORTRAN –ohjelmointikielisinä, noudattaen (ANSI –Fortran 77 tai -90) standardeja. CTPCMAN –käyttöliittymä on tehty C –ohjelmointikielisenä. Fortran ohjelmointikieliset CT –laskentaohjelmat toimivat C –ohjelmointikielisessä käyttöliittymässä.

CTPCMAN -käyttöliittymä sisältää kaikki satamanosturi puolen CT –laskentaohjelmat, projektit ja nosturityypit. Näiden tietojen avulla on helpompi hakea oikea laskentaohjelma tiettyyn nosturiprojektiin. Kun tiedetään nosturityyppi, niin ohjelma antaa nosturia vastaavat laskentaohjelmat ja valmiit lähtöarvot, joista voi avata sen minkä kuormitusta haluaa laskea.

4.2.1 Käyttöliittymän kulkukaavio

CTPCMAN -käyttöliittymään täytyi myös tehdä ohjelman kulkukaavio (KUVIO 3). Eli miten CT -laskentaohjelmat avataan CTPCMAN -käyttöliittymän kautta (liite 2). CT -laskentaohjelmat oli tehty korvaamaan edeltäjänsä, suurin osa kaikista yli 200 laskentaohjelmasta oli jäänyt tarpeettomaksi tai se oli korvattu toisella menetelmällä, esim. CT – laskentaohjelmalla.

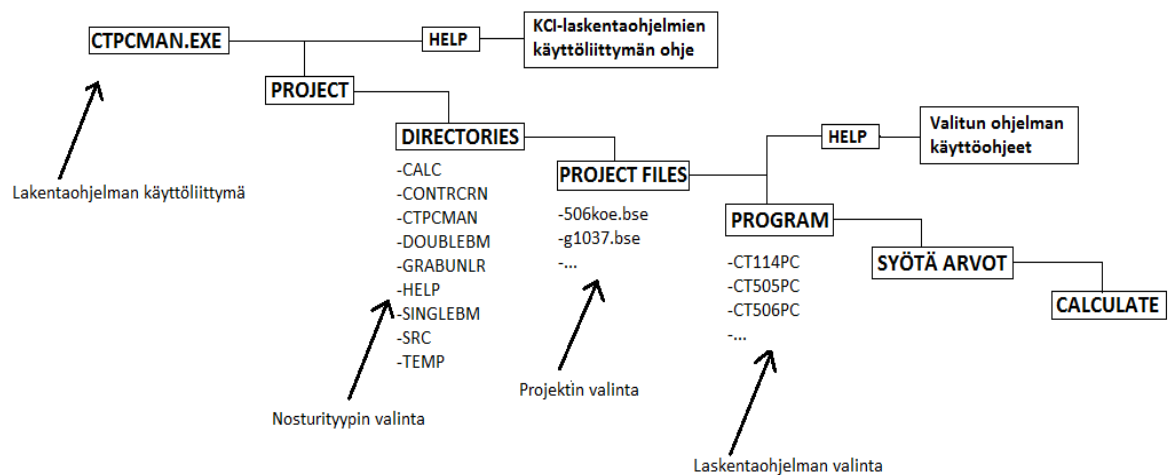
4.2.2 Toimintaperiaate

CTPCMAN –käyttöliittymässä on yksi ikkuna, vähän normaalin Windows-ikkunan tapainen. Ohjelman avaamisen jälkeen nähdään suoraan ikkunan yläruudussa valikot, mistä voi valita seuraavan toimenpiteen. Valikoissa on tarjolla muun muassa Project-niminen valikko. Siitä pääsee valitsemaan haluamansa nosturityypin. Nosturityypin valinnan jälkeen valitaan nosturityyppiä vastaava uusi nosturiprojekti tiedostovalikosta. Käyttöliittymä on tehty tavalla, jossa projektin valinnasta valitaan tietty nosturitiedosto, eli käyttöliittymä hakee etsimällä alihakemistoista kaikki BSE ekstensiolla olevat tiedostot. Ohjelma hakee kaikki nosturityyppiä vastaavat tiedot.

Projektin valinnan jälkeen saadaan valittua nosturityyppiä vastaava laskentaohjelma pudotusvalikosta nimeltä Program. Vain nosturityyppiä vastaavat laskentaohjelmat kuuluvat listaan. Ohjelman valinnassa näkyy ohjelman nimi ja mitä ohjelma laskee. Kun ohjelma on valittu, näkyviin ilmestyy ohjelman lähtöarvotaulukko laskemista varten (kuva 7). Lähtöarvot syötetään ohjelmalle oikeisiin riveihin, jokaisen rivin kohdalla lukee selvästi mitä arvoa ohjelma pyytää.

Käyttöliittymä tarkistaa myös mikäli lähtöarvoihin liittyy erillisiä ns. komponenttitiedostoja. Jos liittyy, niin ohjelma antaa lähtöarvoriviltä alavetovalikon, josta voi antaa valmiita lähtöarvoja. Käyttöliittymä tarkistaa myös mikäli laskentaohjelmassa on tietyt raja-arvot lähtöarvojen syöttämisessä. Jos annettu lähtöarvo ei ole tiettyjen raja-arvojen sisällä, niin ohjelma ilmoittaa siitä. Käyttäjän on kuitattava viesti, ennen kuin ohjelma voi jatkaa.

Laskennan tulos saadaan esiin lähtöarvojen syöttämisen jälkeen Calculate -valintanäppäintä painamalla. Näppäin tallentaa laskelmaohjelman tarvitsemat alkuarvot ja alkuarvotunnuksen erilliseen tilapäistiedostoon. Sen jälkeen ohjelma käynnistää laskentaohjelman, joka lukee tämän tilapäistiedoston ja tallentaa laskelmaohjelman tekemät formatoidut tulokset toiseen tiedostoon. Sitten käyttöliittymä näyttää tiedoston sisällön kuvaruudun tulostusalueessa eli laskennan tulokset. Alla olevassa kuviossa (liite 1) kuvataan miten CT –laskentaohjelma avataan CTPCMAN –ohjelman kautta.



KUVIO 3. CTPCMAN –laskentaohjelman toimintakaavio

Muita valikoita käyttöliittymässä ovat:

- Save, joka tallentaa kaikki alkuarvot nosturitiedostoon.
- Print, joka tulostaa laskentaohjelman tekemän tulostustiedoston Windowsin oletuskirjoittimelle.
- Exit, joka lopettaa käyttöliittymäohjelman ja poistuu Windowsiin.
- Help, joka näyttää käyttöliittymän käyttöohjeen tai valitun laskentaohjelman käyttöohjeen.

CTPCMAN - Project: 506koe - Program: CT505PC

Project... Program... Calculate Save... Print Exit Help

001	Project name.....	=	506koe	
002	User name.....	=	tr	
003	Main hoist max load / outreach....Qm/Um [t,m] =		15	8
004	Main hoist max outreach / load....U1/Q1 [m,t] =		21	8
005	Main hoist min outreach.....U2 [m] =		8	
006	Intermediate outreach / load.....U3/Q3 [m,t] =		38	80
029	Base structure max / med inclinations...[dgr] =		0	0
031	Altitude above sea level.....ALT [m] =		4000	
032	Mean operating wind pressure.....WLN [Pa] =		80	
033	Maximum operating wind pressure.....WLO [Pa] =		250	
034	Storm wind pressure.....WLS [Pa] =		1100	
044	Slewing speed.....Us [rpm] =		1	
053	Slewing acceleration time.....TA [s] =		3	
081	Length of boom.....A [m] =		26.7	
082	Boom hinge location.....R [m] =		0.5	
085	Location of hoist pulley in boom tip...H21 [m] =		0.1	

KUVA 7. CT –laskentaohjelma käynnissä CTPCMAN –käyttöliittymällä

5 TOTEUTUSVAIHTOEHDOT

5.1 Ohjelmien priorisointi

Laskentaohjelmien selvitystyön jälkeen pidettiin eri osastoryhmien kanssa palavereita, joissa priorisoitiin ja listattiin tarpeellisimmat ohjelmat. Työn tarkoituksena ei siis ollut päivittää kaikkia yli 200 laskentaohjelmaa vaan tarpeellisimmat ja käytetyimmät ohjelmat. Tämä tiivistetty lista saatiin juuri priorisoinnin avulla. Ohjelmia jäi priorisoinnin jälkeen jäljelle noin 30 kappaletta ja ne on tarkoitus päivittää toimiviksi 64-bittisillä käyttöjärjestelmillä.

5.2 Menetelmävaihtoehdot

Opinnäytetyössä on tarkoituksena päivittää ainakin osa näistä tarpeellisista CTPCMAN –laskentaohjelmista ja joitakin yksittäisiä, muita tarpeellisia laskentaohjelmia. Päivitysmenetelminä tarkasteltiin eri vaihtoehtoja. Niitä olivat mm. kääntäminen uudella kääntäjällä toimivaksi 64 –bittisessä järjestelmässä, päivitys Excel-tiedostoiksi tai hankkimalla sopivan emulaattorin, jolla ohjelmat toimisivat nykyisissä käyttöjärjestelmissä.

5.2.1 Ohjelmisto kääntäjä

Ohjelmisto-kääntäjän hankkiminen omaan yritykseen olisi vähiten työtä vaativa ratkaisu emulaattorin jälkeen. Ohjelmat toimisivat 64 –bittisessä käyttöjärjestelmässä, mutta tulevaisuuden kannalta se ei olisi välttämättä pitkäaikainen ratkaisu, jos Windowsin käyttöjärjestelmät uudistuisivat merkittävästi. Kääntäjällä käyttöliittymä ei uudistuisi millään tavalla vaan pysyisi samanlaisena. Mahdollisesti ohjelmien kääntämisen jälkeen siirryttäisiin muuntamaan ohjelmia Excel-tiedostoiksi, että ne toimisivat tulevaisuudessakin uusien Windowsien kanssa ja samalla käyttöliittymä uudistuisi.

5.2.2 Excel

Ohjelmien muuntaminen Excel -tiedostoiksi olisi kannattavaa tulevaisuuden kannalta. Ne toimisivat parhaiten ja helpoiten tulevissakin Windows-järjestelmissä, mikäli mahdollisesti tulisi uusi käyttöjärjestelmä tai vastaava. Excelissä myös näkyisivät laskenta-kaavat, millä tulokset saadaan. CT -ohjelmissa niitä ei näy, vaan ohjelmaan syötetään vain arvot ja ohjelma laskee tulokset. Yhteen Excel-tiedostoon voisi myös helposti laittaa vaikka koko nosturityyppiä vastaavat laskentaohjelmat. Jokainen laskentaohjelma olisi Excelissä omalla välilehdellä, josta olisi helppo etsiä tarvittava ohjelma. Excel-ohjelmissa huonona puolena olisi taas päivitystyön määrä. On erittäin hidasta muuntaa Fortran- ja C-kielestä ohjelmaa Exceliin ja varsinkin kun kyseessä on noin 20-30 eri laskentaohjelmaa. Yhden ohjelman muuntaminen kestäisi suunnilleen muutamia kuukausia henkilöltä. Vaihtoehtona olisi työn alihankinta, jos saataisiin ulkoinen ohjelmointiyritys tekemään kaikkien ohjelmien muunnostyö Excel-pohjaisiksi.

5.2.3 Emulaattori

Kolmas vaihtoehto olisi sopivan emulaattorin hankkiminen. Sen avulla ohjelmat voisivat toimia 64-bittisissä käyttöjärjestelmissä. Emulaattorin hankkiminen olisi vain väliaikainen ratkaisu, koska sen toiminnan takaaminen on vaikeaa. Se voi toimia tai sitten se ei emuloi ohjelmaa kunnolla ja toimintakyky on sen vuoksi heikko. Emulaattorin asentaminenkin jokaiselle koneelle olisi myös aika työlästä.

Opinnäytetyön aikana kokeiltiin ajaa laskentaohjelmia DOSBox-emulaattorilla. Se on ilmainen Internetistä ladattavissa oleva ohjelma. Ohjelmat eivät aluksi toimineet DOSBox:ssa suoraan puutteellisen lähdekoodin takia. Tarvittavien lisäyksien jälkeen laskentaohjelmia pystyi ajamaan DOSBox:n kautta 64-bittisessä Windows-käyttöjärjestelmässä. Ohjelmien ajo tuntui vähän kankealta tällä tavalla, koska niiden käyttöliittymä pitää avata aina DOSBox:n kautta. Emulaattorin toiminnastakaan ei ole mitään varmuuksia, eli se saattaa kaataa ohjelmat milloin vain tai ohjelmoi niitä puutteellisesti.

5.2.4 KC -laskentaohjelmat

Neljäs vaihtoehto olisi mahdollisesti Konecranesin omien KC –laskentaohjelmien (liite 2) käyttö. Ne voisivat korvata CT -laskentaohjelmat ja osa niistä nykyään laskeekin samoja tietoja kuin CT -laskentaohjelmat. Ongelmaksi muodostuu tässä vaihtoehdossa se, etteivät satamanosturipuolen CT –ohjelmat ja KC –ohjelmat mene aivan yksi yhteen, eli eivät laske ihan samaa tietoa samantyyppiselle nosturille. Eli esimerkiksi CT –ohjelmat laskevat tietoa satamanostureille ja KC –ohjelmat toisille nostureille. Vain osa päivitettävistä CT -laskentaohjelmista vastaa sisällöltään KC- ohjelmia, jolloin kaikkien ohjelmien ohjelmointityömäärä saattaisi olla liian suuri. Periaatteessa voitaisiin ohjelmoida vain ne ohjelmat, jotka laskevat samoja asioita kuin KC- ohjelmat ja päivittää loput muilla keinoilla. Tässäkin tapauksessa täytyy selvittää minkälainen päivitystyö olisi korvata vanhat ohjelmat KC –laskentaohjelmilla vai olisiko se liian työläs ratkaisu verrattuna muihin. Hyvä asia KC –ohjelmissä on se, että ne toimivat 64-bittisissä käyttöjärjestelmissä ja laskentaohjelmien käyttöliittymä paranisi. Tätä voidaan pitää jatkokehitysmahdollisuutena, sitten kun CT -ohjelmat ovat saatu päivitettyä toimimaan 64-bittisessä helpommalla keinolla.

The screenshot shows the KC Rope 1.7.1 software interface. The main window is titled "KC Rope 1.7.1" and has a menu bar with "File", "Help", and "Online". Below the menu bar is a toolbar with "File", "Help", and "Online" buttons. The main area is divided into several sections:

- Top Bar:** "Calculated ropes" dropdown showing "11 8x19S LHL".
- Navigation Tabs:** "Summary", "Results of loadings", "Test Criteria".
- Left Panel:**
 - Preselected ropes:** "None" button, "Calculate" button, "Include rejected ones" checkbox.
 - Calculate according to:** "ISO/FEM/JIS" dropdown.
 - ISO/FEM/JIS:**
 - Duty group:** "M4" dropdown.
 - Rated working load of appliance:** "7000" kg.
 - Mass of the pulley block and/or another lifting attachments:** "200" kg.
 - Number of falls:** "4" (-).
 - Efficiency of a single sheave:** ".99" (-).
 - Efficiency of the whole reeving system:** "0.97" (-).
 - Rope inclination with respect to the drum axis:** "10" de.
 - Stationary:** checkbox.
 - Predefined components of the rope system:**
 - Pitch circle diameter of the drum:** "300" mm.
 - Pitch circle diameter of the smallest sheave:** "300" mm.
 - Pitch circle diameter of the compensating sheave:** "200" mm.
 - Minimum used groove radius:** "6,7" mm.
 - Maximum used groove radius:** "6,7" mm.
 - Minimum used depth of the groove:** "22" mm.
 - Minimum used groove angle:** "40" de.
 - Maximum used groove angle:** "45" de.
- Right Panel:**
 - Calculated loadings:** "Loading 1: According to: ISO/FEM/JIS" dropdown.
 - Results (1) - ISO 4308 / FEM / JIS C 9620:**
 - Empirical factor for minimum breaking load, K':** "0,346" (-).
 - Minimum practical coefficient of utilization, Zp:** "4" (-).
 - Minimum tensile strength of the rope wire, R0:** "2160" MP.
 - Rope selection factor (minimum), C:** "0,0732" $\frac{1}{\sqrt{N}}$.
 - Formula:** $C = \sqrt{\frac{Z_p}{K' \cdot R_0}}$
 - Results (2) - ISO 4308 / FEM / JIS C 9620:**
 - Maximum rope tension, S:** "18,7" kN.
 - Minimum required breaking force, Fmin:** "74,7" kN.
 - Minimum breaking force of the rope:** "126" kN.
 - Formula:** $F_{min} = S \cdot Z_p$
 - Minimum allowed rope diameter, dmin:** "10" mm.
 - Maximum allowed rope diameter, dmax:** "12,5" mm.
 - Formula:** $d_{max} = 1,25 \cdot d_{min}$
 - Selection factor for the drum diameter, h1:** "16" (-).
 - Selection factor for the sheave diameter, h2:** "18" (-).
 - Selection factor for the comp. sheave diam., h3:** "14" (-).
 - Rope type factor, t:** "1" (-).
 - Minimum pitch circle diameter of the drum, D1:** "160" mm.
 - Minimum pitch circle diameter of the sheave, D2:** "180" mm.
 - Min. pitch circle diam. of the comp. sheave, D3:** "140" mm.
 - Formulas:**
 - $D_1 \geq h_1 \cdot t \cdot d_{min}$
 - $D_2 \geq h_2 \cdot t \cdot d_{min}$
 - $D_3 \geq h_3 \cdot t \cdot d_{min}$

KUVA 8. KC –laskentaohjelman käyttöliittymä

5.2.5 Päivitystyön alihankinta

Päivitystyön voisi myös antaa tehtäväksi jollekin osaavalle ohjelmointiyritykselle. Alan ammattilaiselta työ hoituisi mahdollisesti nopeammin kuin oman organisaation jäseneltä, jolla mahdollisesti olisi muita projekteja samaan aikaan. Päivitysprojekti voisi hidastua, jos työntekijän kiireisempi projekti vie suurimman osan ajasta. Huonona puolena ulkopuolisen työvoiman käytössä olisi mahdollisesti tiedon puute. Ohjelmoija ei välttämättä ymmärtäisi laskentojen tulosten oikeellisuutta.

Laskentaohjelmien käyttöliittymään ei suunnitelmien mukaan olisi luvassa suurempia muutoksia, koska ohjelmat toimivat oikein jo 32-bittisessä käyttöjärjestelmässä. Tarvitisi vain saada laskentaohjelmien käyttöliittymä toimivaksi 64-bittisessä käyttöjärjestelmässä. Muutama korjaus käyttöliittymään olisi tulossa, mutta niiden ei uskota tuovan ongelmia päivitystyöhön. Korjauksia olisi valikkonäppäimissä, jotka eivät toimi nykyisessä käyttöliittymässä halutulla tavalla.

6 TOTEUTUSVAIHTOEHDON VALINTA

Päivitysmenetelmän valintapäätös tehtiin sen perusteella, mikä olisi helpoin ja kätevin tapa päivittää kaikki tarvittavat laskentaohjelmat kerralla. Tarjolla olisi toki ollut muutamia muita menetelmiä, joissa osa ohjelmista olisi korvattu toisilla ohjelmilla ja osa taas toisilla. Tämä olisi ollut mahdollista joidenkin ohjelmien samankaltaisuuden takia. Konecranesilla olisi ollut käytössä eri osastoilla kuin satamanosturipuolella samankaltaisia laskentaohjelmia, mutta vain erityyppisille nostureille tarkoitettuja. Näin olisi saatu osa laskentaohjelmista päivitettyä periaatteessa pienellä ohjelmoinnilla. Näin ei kuitenkaan haluttu tehdä, koska näin vain osa ohjelmista olisi tullut päivitettyksi ja loput olisi täytynyt päivittää jollain muulla menetelmällä.

Menetelmän päätökseen vaikuttivat myös tulevaisuuden näkymät, eli toimisiko ohjelma myös tulevaisuudessa ilman samanlaista mittavaa päivitystyötä. Tämän jälkeen Excel-tiedostoiksi muuntaminen olisi ollut melkein pä paras ratkaisu, koska se olisi ollut helppo siirtää aina uusista Excel-versioista uusiin. Ongelmana on vain suuri työmäärä, jossa mahdollisesti menisi monta työkuukautta monelta ihmiseltä. Ongelmaksi voisi myös muodostua laskentojen tulosten oikeellisuus. Mikäli laskentaohjelma siirretään Excel-pohjaan, niin kaavojen ja laskentojen tuloksien on oltava oikeita. Tämä taas tarkoittaa sitä, että työn tekijän pitää osata ohjelmoinnin lisäksi tuntea laskentaohjelman tarkoitus ja kaavat.

Menetelmien arvioimiseksi tehtiin pisteytystaulukko (TAULUKKO 3). Arvosanoja menetelmille annettiin ominaisuudesta riippuen 0 – 3 pisteeseen.

TAULUKKO 3. Päivitysmenetelmien pisteytys

Ominaisuus	Excel	Kääntäjä	Emulaattori	KC-ohjelma	Alihankinta
Asennettavuus	3	2	2	2	3
Työaika	0	2	3	2	3
Työmäärä	1	2	3	2	3
Kestävyys	3	2	1	3	2
Kustannus	1	2	3	2	1
Käyttö	3	2	3	3	3
Luotettavuus	2	3	2	2	2
Pisteet	13	15	17	16	17

6.1 Päivitystyön suorittajan valinta

Päivitystyöntekijää mietittiin myös muutamista vaihtoehdoista. Vaihtoehtoja työn tekemiseen olivat joko työn tekijöiden etsiminen omasta yrityksestä tai sitten alihankkia työ joltain ohjelmointiyritykseltä. Vaihtoehtoja vertaillen nousi esiin muutamia kysymyksiä kummastakin vaihtoehdosta.

Työntekijän pitäisi osata ohjelmoida tämän tyyppisiä laskentaohjelmia sekä tuntea ohjelman laskukaavoja ja ymmärtää vähän mitä ne laskevat. Tulosten pitäisi uusissa ohjelmissa vastata oikeita, muuten päivitettävillä ohjelmilla ei tekisi mitään, koska niiden tuloksiin ei voisi luottaa. Aika oli myös ratkaiseva tekijä tässä päätöksessä.

Kuinka kauan työn tekeminen kestäisi oman organisaation jäseneltä ja olisiko hänellä myös muita töitä tämän ohessa, että päivitystyö mahdollisesti voisi viivästyä kiireisempien töiden edessä. Siksi työn alihankkiminen olisi parempi ratkaisu, koska erillinen ohjelmointiyritys ainakin keskittyisi vain tämän päivitystyön tekemiseen. Huonoina puolina alihankinnassa voisi pitää mahdollisesti suuria kustannuksia työlle ja työntekijän kokemuksen puutetta tuntea ohjelmissa laskettavat asiat.

Tärkeimmäksi tekijäksi päivitystyössä nostettiin aika. Työ saataisiin tehtyä mahdollisimman nopeasti ja laskentaohjelmat toimimaan. Työn tekemiseen soveltuvia ohjelmointiyrityksiä etsittiin aluksi Internetistä. Selvitettiin myös olisiko Konecranesin muissa yksiköissä yhteyksiä johonkin ohjelmointi yritykseen, josta asiaa voisi kysellä. Konecranesin laskentainsinööriltä selvisi, että heidän puolellaan oli yhteyksiä tiettyyn ohjelmointiyritykseen Riihimäellä. Yritys oli nimeltään Riihisoft Oy ja yritys teki samaan aikaan yhteistyötä laskentainsinöörien oman projektin kanssa.

Riihisoftilta selviteltiin olisivatko he kiinnostuneita tästä päivitystyöstä ja millainen olisi mahdollinen työmäärä ja aikataulu. Myös muilta yrityksiltä kyseltiin, mikäli tällainen työ kiinnostaisi.

Hyvinkäältä löytyi NSD niminen yritys, jolta myös kysyttiin työn tekemistä. Konecranes oli ennenkin tehnyt projekteja yhtiön kanssa. Yrityksestä kerrottiin, että tällainen käännöstyö onnistuisi heiltä. Päätettiin pitää NDS:n edustajan kanssa palaveri aiheesta, jossa hänelle näytettiin joitakin ohjelmia ja selvitettiin miten ne toimivat.

6.2 Päivitysratkaisu

Päivitystyön suorittajaksi päätettiin valita NSD ohjelmointiyritys. NSD:n edustajan ehdotus oli, että aluksi käännettäisiin C –ohjelmointikielisen laskentaohjelmien käyttöliittymän toimivaksi 64 bittiselle käyttöjärjestelmälle. Tämän jälkeen käännettäisiin Fortran ohjelmointikielisiä laskentaohjelmia 1-3 kappaletta, jotta nähtäisiin alkaisivatko ne toimia halutulla tavalla uudessa 64-bittisessä ympäristössä. Mikäli muutamien ohjelmien käännöstyö onnistuu ja ohjelmat toimivat, niin voidaan sopia jäljelle jäävien laskentaohjelmien käännöstyöstä.

Konecranesin edustajalla oli päätöksen teon yhteydessä samanlainen näkemys laskentaohjelmien päivittämispolusta. Ensin käännettäisiin vain muutamia ohjelmia C# -ohjelmointikielelle, jonka jälkeen katsottaisiin minkälainen työmäärä ja aikataulu niissä on. Sen jälkeen voitaisiin arvioida lopullinen työmäärä ja aikataulu, jos ohjelmat toimivat kunnolla. Alustava aikataulu 1-3 ohjelman kääntämistyölle oli 10 työpäivää. Tänä aikana käännetään muutama laskentaohjelma, jolloin saadaan kokonaiskuva kaikkien laskentaohjelmien päivitystyömäärästä.

Ohjelmat saatiin siis toimimaan DOSBox emulaattorilla 64 –bittisessä Windowsissa. Tätä voidaan pitää ensiaskeleena työn onnistumisessa. Jatkokehitystyönä tästä on siis käännettää käyttöliittymän varsinaiset laskentamodulit uudelle C# -ohjelmointikielelle ja sitä kautta lähteä parantamaan ohjelmien käyttöliittymää yhdessä NSD:n kanssa. Suunnitelmissa on mahdollisesti tehdä ohjelmille yksinkertainen käyttöliittymä ja tämän jälkeen suunniteltaisiin ohjelmien sisällyttämistä Konecranesin KC -laskentaohjelmiin. Tarkka päivitystyön kuvaus löytyy liitteestä 3.

7 POHDINTA JA JATKOKEHITYSMAHDOLLISUUDET

7.1 Pohdinta

Työn tarkoituksena oli tutkia yrityksen satamanosturipuolen laskentaohjelmien päivitysmahdollisuuksia ja siitä saavutettavaa hyötyä. Tutkintotyössä tarkasteltiin erilaisia menetelmiä ja tapoja, joilla laskentaohjelmia voitaisiin kehittää ja päivittää. Tavoitteena oli valita päivitystyöhön vain tarpeellisimmat ja tärkeimmät laskentaohjelmat uudelle käyttöjärjestelmälle toimivaksi.

Työssä esitettyjen laskentaohjelmien päivitystyötä uudelle käyttöjärjestelmälle voidaan pitää merkittävä työnä satamanosturiosastolle suunnittelutyön helpottamiseksi. Päivityksen jälkeen laskentaohjelmat toimivat jokaisessa tietokoneessa kunnolla, myös 32-että 64-bittisissä järjestelmissä. Päivitystyöhön kuluva aika voi olla muutamia kuukausia joten lopullisia tuloksia ei päästä arvioimaan ennen tämän opinnäytetyön valmistumista.

Opinnäytetyölle asetettujen tavoitteiden voidaan todeta onnistuneen tavoitellun päivitystyömenetelmän löytämisen myötä ja laskentaohjelmien toimintaan saamisen seurauksena. Työssä määriteltiin miten päivitystyö lopulta tehdään ja kenen toimesta monien eri vaihtoehtojen vertailun jälkeen. Työn tavoitteena pidettiin parhaan mahdollisen päivitysmenetelmän löytämistä nopeassa ajassa ja siinä onnistuttiin hyvin. Päivitystyöntekijän valinnassa tehtiin myös paljon työtä, että saatiin oikeanlainen yritys työhön mukaan. Aika ja kustannukset olivat tässä päätöksessä tietysti tärkeitä asioita. Työn yhtenä tavoitteena oli myös saada se kokonaisuudessaan valmiiksi eli myös itse päivitystyö tehtyä opinnäytetyön ohessa, mutta tähän tavoitteeseen ei päästy työn laajuuden ja keston vuoksi. Työ siis päätettiin tehdä ulkoisen yrityksen kanssa yhteistyössä, joten neuvottelut työn tekemisen suhteen kestivät molemmilta osapuolilta, ennen kuin päästiin sopimukseen työnteon aloittamisesta ja lopullisesta työmäärästä.

7.2 Jatkokehitysmahdollisuudet

Työn jatkokehitysmahdollisuudet ovat mahdollisesti laskentaohjelmien käyttöliittymien uudistamisessa. Laskentaohjelmat olisi tulevaisuudessa hyvä saada kaikki samaan käyttöliittymään sisältyviksi. Nykyään ne ovat siis eri käyttöliittymissä, esim. Excel, CTPCMAN ja QB.

Konecranesilta päätettiin, että tämän C# -ohjelmointikielelle kääntämisen jälkeen tehtäisiin kokonaan uusi yksinkertainen graafinen käyttöliittymä ohjelmille. Käyttöliittymän uudistamisella ohjelmista saataisiin helppokäyttöisempi ja selkeämpiä. Vanhat ohjelmat eivät ole niin selkeitä ensikertalaiselle käyttäjälle ja niitä on melko hidas käyttää. Tämän jälkeen ohjelmia tai uutta käyttöliittymää voitaisiin alkaa sisällyttämään Konecranesin omaan KC –laskentaohjelmistoperheeseen.

Toisena vaihtoehtona voidaan pitää laskentaohjelmien siirtämistä Excel-työkirjaan. Käyttöliittymä uudistuisi ja ohjelmat saataisiin samaan käyttöliittymään sisältyviksi. Excelissä on vanhoja ohjelmia selkeämpi käyttöliittymä. Laskujen kaavat, joista tulokset tulevat, näkyisivät selvästi. Ohjelmia olisi helpompi kehittää ilman koodikieltä ja mahdolliset laskentojen virheet huomattaisiin myös helpommin. Tällä hetkellä työskennellään kuitenkin vielä uuden käyttöliittymän kanssa.

LÄHTEET

Konecranes Oy. 2012. Yrityksen verkkosivut.

Ojapalo, E. Esimies, Vihottula, M. Koneistoinsinööri, Rautajärvi, H. Koneistoinsinööri, Konecranes Oy. 2013. Haastattelu/Palaveri 06.02.2013. Haastattelija Lahti, J. Hyvinkää.

Ojapalo, E. Esimies, Tuomainen, J. Rakenneinsinööri, Räsänen, T. Rakenneinsinööri, Konecranes Oy. 2013. Haastattelu/Palaveri 04.02.2013. Haastattelija Lahti, J. Hyvinkää.

Wikipedia.org/Ohjelmointikielet

Beginning Programming, Kingsley-Hughes & Kingsley-Hughes, 2005.

TIOBE Programming Community Index, 2013.

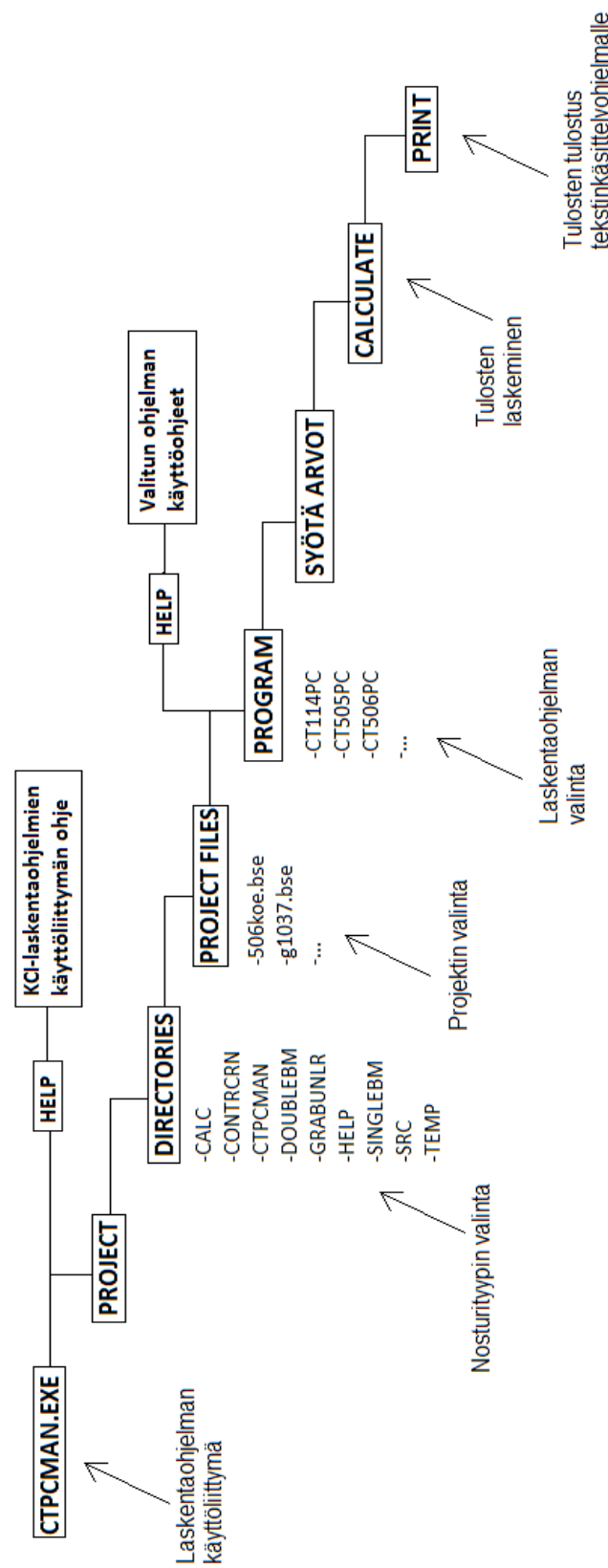
<http://www.tiobe.com/index.php/content/paperinfo/tpci/index.html>. Luettu 28.2.2013

Joentakanen, K. 2010. Pygame-peliohjelmointi. Tietotekniikan koulutusohjelma. Keski-Pohjanmaan ammattikorkeakoulu. Opinnäytetyö.

Ojapalo, E. Esimies, Lappalainen, J. Tutkimus- ja kehitysinsinööri, Aaltonen, T. Esimies, laskenta. Konecranes Oy. 2013. Haastattelu/Palaveri 05.04.2013. Haastattelija Lahti, J. Hyvinkää.

LIITTEET

Liite 1. CTPCMAN –käyttöliittymän kulkukaavio



Liite 2. Esimerkki CTPCMAN –käyttöliittymän lähdekoodista.

```
// Projekti:          CTPCMAN
// Projektikuvaus:    Käyttöliittymä nosturilaskentaohjelmille
//
// Moduulin nimi:     compnet.h
// Kuvaus:             Komponentti tiedoston käsittely
//
// Versiohistoria:
// 19.5.1197   v1.0.0   Luotu
//
// *****

#include "precomp.h"
#pragma hdrstop

#include "compnent.h"
#include "util.h"
#include <stdlib.h>
#include <dir.h>
#include "ctpcman.rh"

int TCompButton::focusId = 0;
TFont* ComponentDialog::ListBoxFont = 0;
TFont* CompListDialog::ListBoxFont = 0;
// Millä rivillä olevaa nappia painettiin viimeksi

// napin käsittelemät viestit
DEFINE_RESPONSE_TABLE1(TCompButton, TButton)
    EV_WM_LBUTTONDOWN,
END_RESPONSE_TABLE;

// *****
//
// -----
// Kuvaus:
// Parametrit:
// Palauttaa:
// *****
TCompButton::TCompButton(int paramId, Twindow* parent, int winId)
    : TButton(parent, winId, "...", 0, 0, 3, 1), btnId(paramId)
{
}

// *****
//
// -----
// Kuvaus:
// Parametrit:
// Palauttaa:
// *****
TCompButton::~TCompButton()
{
}

// *****
//
// -----
// Kuvaus:
// Parametrit:
// Palauttaa:
// *****
void TCompButton::EVLButtonDown(uint /*modKeys*/, TPoint& /*point*/)
{
    focusId = btnId;

    // komponentti nappia painettu
    Parent->SendNotification(Attr.Id, CN_COMPONENT_BTN, Hwindow);
}

// *****
//
// -----
// Kuvaus:
// Parametrit:
// Palauttaa:
// *****
```

Liite 3. Laskentaohjelmien päivityssuunnitelma (NSD)

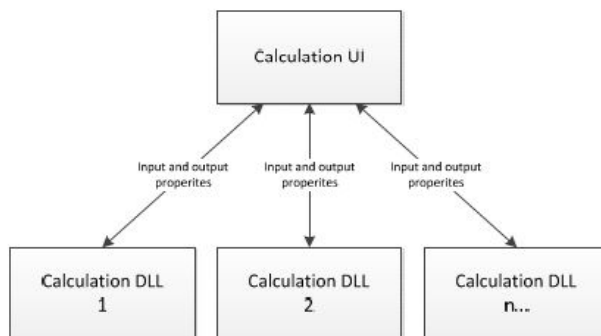
1(3)

KONECRANES – Ports laskentaohjelmat

17.4.2013 / nwe

Vaihe-1: Fortran – C# konvertointi**Yleistä**

- Edetään yksi laskentaohjelma kerrallaan
- Fortran ohjelma koodataan C# kielellä 64bit DLL luokaksi
- DLL luokat tehdään siten, että niitä voidaan myöhemmässä vaiheissa käyttää muiden KCI laskentaohjelmien ja konfiguraattorien toimesta
- Kieli on englanti (käyttöliittymä, muuttujat, propertyt, kommentit ym.)
- Käyttöliittymä ja DLL luokat ovat projekteja samassa solutionissa

**Käyttöliittymä**

- Tekniikka: WPF, 64bit EXE
- Käyttöliittymässä valitaan ensin laskelman tyyppi, jonka perusteella määräytyvät input kentät
- Käyttöliittymässä on kaksi tabia: Input ja Output
- Käyttöliittymän Calculate-painike
 - Muodostaa laskennan DLL-luokasta objektin
 - Määrittää objektille input propertyt
 - Kutsuu Calculate-metodia
 - Hakee yhdestä OutputText propertystä laskennan tulokset
 - Tuo tulokset Output-tabille
 - Lisäksi oltava ErrorDescription output property, joka on tyhjä, mikäli laskennassa ei ilmene virheitä. Muussa tapauksessa sisältää virheen selityksen
- Talletus suoritetaan serialisoimalla laskentaluokka XML muotoon, kun laskentaluokka sisältää input- ja output-propertyt
- Ohjelmisto voidaan ottaa tuotantokäyttöön

Liite 3. Laskentaohjelmien päivityssuunnitelma (NSD)

2(3)

Esimerkkikäyttöliittymä

CalcUISample - NC109 - Heavy duty motor calculation [MyCalculation.xml]

File Calculate Options

Input Output

Max load t

Span m

Height of lift m

Duty class

Etc.

Etc.

Etc.

Etc.

Calculate

Idle

The screenshot shows a Java Swing window titled "CalcUISample - NC109 - Heavy duty motor calculation [MyCalculation.xml]". The window has a menu bar with "File", "Calculate", and "Options". Below the menu bar is a toolbar with four icons. There are two tabs, "Input" and "Output", with "Input" currently selected. The main content area displays the text "Calculation started 2013-04-11 12:10:00" followed by a list of parameters and their values:

Lozem:	12.00 m
Ipsum:	121.00 mm
Dolor:	DDV
Consectetur:	Upper
jne..	
jne..	
jne..	
jne..	
jne..	
jne..	
jne..	

At the bottom of the window, there is a status bar with the text "Idle" and a small progress indicator.

Vaihe-2: KCI yleinen yhteensopivuus

- DLL luokkien output propertyt (tai luokat) määritetään yhdessä IT:n kanssa
- Määritetyt output-propertyt koodataan laskentaluokkiin
- Samalla tarkastetaan, että myös input propertytien nimet ovat yleisen standardin mukaiset

Vaihe-3: Tietokannat käyttöön

- Tarvittavalta osin otetaan käyttöön olemassaolevat tietokannat esim. moottoritietojen osalta ja luodaan tarvittavat uudet taulut
- Otetaan käyttöön IT:n oliokanta-moduli, jolla suoritetaan laskelmien tallennus ja haku tietokantaan

Vaihe-4: Käyttöliittymän jatkokehitys

- Output tietojen osalta toteutetaan erilliset kentät, ei vain yksi iso textbox
- Huomioidaan eri maiden käyttämät mittayksiköt, myös input kentissä
- Huomioidaan mahdolliset kenttien väliset relaatiot ja raja-arvot sekä niiden tarkastukset

Vaihe-5: Automaattinen testaus

- DLL-luokille toteutetaan yksikkötestit DLL-kutsutasolla

Aikatauluestimaatti (vaihe-1)

- n. 1 tvk (5 tpv) laskentamodulin Fortran koodin käännökselle C#:iin
- n. 1 tvk käyttöliittymän toteutukseen, kokonaisuuden testaukseen ja mahdollisten virheiden korjaukseen
- Yhteensä n. 2 tvk / laskentamoduli